CLAIMS

What is claimed is:

1.   A method comprising:

2   providing prerequisite information regarding pages of a graphical user interface

3   (GUI) that are prerequisites to other pages of the GUI, each page including one or more

4   sub-components;

5   in response to a request to display a destination page and with reference to the

6   prerequisite information, identifying one or more prerequisite pages associated with the

7   destination page;

8   determining which sub-component of the one or more sub-components of an

9   identified prerequisite page is a decider sub-component that is capable of confirming

10   whether or not requirements of the identified prerequisite page have been satisfied;

11   determining whether or not the requirements of the identified prerequisite page

12   have been satisfied by invoking a method of an instance of the decider sub-component that

13   causes stored information regarding the state of the identified prerequisite page to be

14   retrieved from a current environment/context; and

15   causing the output of the destination page to be displayed if all the requirements of

16   the one or more identified prerequisite pages have been satisfied, otherwise causing the

17   output of a prerequisite page of the one or more identified prerequisite pages having one or

18   more requirements that have not been satisfied to be displayed.

1   2.   The method of claim 1 wherein the prerequisite information is stored in a Java

2   properties file.

1   3.   The method of claim 1, further comprising supporting hierarchical relationships of

2   prerequisite pages by:

3   iterating through each of the identified prerequisite pages associated with the

4   destination page in a predetermined order until encountering the first prerequisite page that

5   has one or more requirements that have not been satisfied; and

6   displaying the first prerequisite page of the identified prerequisite pages before

7   displaying a second prerequisite page of the identified prerequisite pages that has one or

8   more requirements that have not been satisfied, the second prerequisite page being

9   dependent upon the first prerequisite page according to the predetermined order.

1   4.   The method of claim 1, wherein the request to display the destination page

2   comprises a HyperText Transfer Protocol (HTTP) request, and wherein the pages of the

3   GUI comprise web pages.

1   5.   The method of claim 1, wherein the prerequisite information includes a prerequisite

2   property for each of the pages of the GUI, the prerequisite property comprising a string

3   identifying the one or more prerequisite pages.

1   6.   The method of claim 1, wherein the prerequisite information is structured as a list

2   of attribute-value pairs, and wherein the syntax for identifying a first page, $page_1$, and a

3   second page, $page_2$, as prerequisites of a third page, $page_3$, is substantially as follows:

4     $page_3.prereq = page_1\ page_2$.

1   7.   The method of claim 1, further comprising modifying the prerequisite information

2   without necessitating recompilation of software code.

1   8.   The method of claim 1, wherein said determining whether or not the requirements

2   of the identified prerequisite page have been satisfied includes requesting that a page

3   prerequisite object verify whether all its requirements have been satisfied.

1   9.   The method of claim 1, wherein page objects corresponding to the pages of the

2   GUI and page prerequisite objects responsible for ensuring satisfaction of one or more

3   prerequisite conditions are loosely coupled and may be dynamically associated with each

4   other by way of the prerequisite information.

1   10.   A graphical user interface (GUI) system for enforcing page prerequisites

2   comprising:

3   a properties data store including information regarding pages of the GUI that are

4   prerequisites to other pages of the GUI;

5   a base agent to respond to requests to display a destination page of the GUI, in

6   response to a request to display the destination page, the base agent causing the output of

7   the destination page to be displayed if all the requirements of one or more prerequisite

8   pages associated with the destination page have been satisfied, otherwise causing the

9   output of a prerequisite page of the one or more prerequisite pages to be displayed; and

10   a prerequisite factory decoupling the pages from their respective prerequisite pages,

11   the prerequisite factory to determine whether or not requirements of one or more identified

12   prerequisite pages have been satisfied by causing information regarding the state of the one

13   or more identified prerequisite pages to be retrieved from a current environment/context.

1   11.   The system of claim 10, wherein the prerequisite factory identifies the one or more

2   prerequisite pages associated with the destination page by accessing the properties data

3   store, determines which of one or more sub-components of an identified prerequisite page

4   is a decider sub-component that is capable of confirming whether or not requirements of

5   the identified prerequisite page have been satisfied, creates an instance of the decider sub-

6   component, and determines whether or not the requirements of the identified prerequisite

7   page have been satisfied by invoking a method of the instance that causes information

8   regarding the state of the identified prerequisite page to be retrieved from the current

9   environment/context.

1   12.   The system of claim 10, wherein the prerequisite information is stored in a Java

2   properties file.

1   13.   The system of claim 10, wherein the prerequisite factory supports hierarchical

2   relationships of prerequisite pages by:

3   iterating through each of the identified prerequisite pages associated with the

4   destination page in a predetermined order until encountering the first prerequisite page that

5   has one or more requirements that have not been satisfied; and

6   displaying the first prerequisite page of the identified prerequisite pages before

7   displaying a second prerequisite page of the identified prerequisite pages that has one or

8   more requirements that have not been satisfied, the second prerequisite page being

9   dependent upon the first prerequisite page according to the predetermined order.

1   14.   The system of claim 10, wherein the requests correspond to HyperText Transfer

2   Protocol (HTTP) requests, and wherein the pages of the GUI comprise web pages.

1   15.   The system of claim 10, wherein the information of the properties data store

2   includes a prerequisite property for each of the pages of the GUI, the prerequisite property

3   comprising a string identifying the one or more prerequisite pages.

1   16.   The system of claim 10, wherein at least a portion of the information of the

2   properties data store is structured as a list of attribute-value pairs, and wherein the syntax

3   for identifying a first page, $page_1$, and a second page, $page_2$, as prerequisites of a third

4   page, $page_3$, is substantially as follows:

5        $page_3$.prereq = $page_1$ $page_2$.

1   17.   The system of claim 10, wherein prerequisite relationships among two or more

2   pages of the pages of the GUI may be modified without necessitating recompilation of

3   software code by editing the information of the properties data store.

1   18.   The system of claim 10, further comprising page objects corresponding to the pages

2   of the GUI and page prerequisite objects responsible for ensuring satisfaction of one or

3   more prerequisite conditions are loosely coupled and may be dynamically associated with

4   each other by way of the prerequisite information.

1   19.   A method comprising:

2   identifying, at run-time, one or more prerequisite web pages associated with a

3   requested web page by accessing a properties file;

4   determining whether requirements of the one or more identified prerequisite pages

5   have been satisfied; and

6   causing the output of the requested web page to be displayed if all the requirements

7   of the one or more identified prerequisite pages have been satisfied, otherwise causing the

8   output of a prerequisite page of the one or more identified prerequisite pages having one or

9   more unsatisfied requirements to be displayed.

1   20.   A machine-readable medium having stored thereon data representing sequences of

2   instructions, the sequences of instructions which, when executed by a processor, cause the

3   processor to:

4   identify one or more prerequisite pages associated with a destination page by

5   accessing a properties file in response to a request for the destination page, the properties

6   file including prerequisite information regarding pages of a graphical user interface (GUI)

7   that are prerequisites to other pages of the GUI;

8   determine which sub-component of an identified prerequisite page is capable of

9   confirming whether or not requirements of the identified prerequisite page have been

10   satisfied;

11   determine whether the requirements of the identified prerequisite page have been

12   satisfied by invoking a method of an instance of the sub-component that causes stored

13   information regarding the state of the identified prerequisite page to be retrieved from a

14   current environment/context; and

15   cause the output of the destination page to be displayed if all the requirements of

16   the one or more identified prerequisite pages have been satisfied, otherwise cause the

17   output of a prerequisite page of the one or more identified prerequisite pages having one or

18   more requirements that have not been satisfied to be displayed.